



## Communication protocol and command format

### 1、 general communication process between PC and Arduino:

- 1) PC initiates communication by sending the “start” command to Arduino
- 2) Once Arduino receives the “start” command, start waiting for the valid data sent from PC
- 3) Arduino will send a request at the right time and return the state.
- 4) PC will send the first row of data in the buffer issued when when the request is received
- 5) When the data is received by Arduino , it will be stored in the the buffer and executed when it fits algorithm schedule.
- 6) A ‘termination’ commend should be send when PC goes offline.

### 2、 command and data pack format

(1) start command: {0xa5, 0x00, 0x00, 0x11, 0x11, 0x22, 0x22, 0x33, 0x33, 0x00, ....., 0x5a};

(2) terimination command : {0xa5,0x44,0x44,0x55,0x55,0x66,0x66,0x77,0x77,0x00, ....., 0x5a};

(3) Arduino request command: {0xa5,0x00....., 0x5a}

header	X	Y	Z	RHead	baseAngle	longArmAngle	shortArmAngle	pawArmAngl	isGrab	tail
0xa5										0x5a

Note:

- In (1)(2), the length of start and termination commend is 42 bytes, bold section of the commend - ‘0x00, .....,’ represents 32 ‘0x00’ in total.
- In (3), Request commend has total length of command is 38 bytes, and 36 bytes(shown in Bold) in the middle of the commend represents 9 float parameters.

(4) the data pack PC send to Arduino is showing as below::

header	state	Axis	X	Y	Z	RHead	isGrab	StartVe	EndVel	MaxVe	tail
0xa5	1-8	0-10									0x5a

- The length of a data pack is 42 bytes, which contains a head(1 byte), 10 float (32 bits= 4 bytes) parameters, and a tail(1 byte).
- Each parameter is sent as bytes and low byte first before high byte.



3、 Table 1 : Parameter explanation of data pack PC send to Arduino

parameter	meaning
State	used to identify each operate ways
Axis	Used to identify deference single movement(shown on table 2) when State is 2 or 7.
X、 Y、 Z	displacement increments
RHead	rotation angle of arm head
isGrab	whether gripper close or whether pump sniff or not.
StartVel、 EndVel and MaxVel	limitation of velocity

Table 2: Axis corresponding operation when single and straight line control.:

state	Axis	=2 Single axis control	=7 Straight line control
0		no bottom pressed	No bottom pressed
1		Axis1+	forward
2		Axis1-	backward
3		Axis2+	turn left
4		Axis2-	turn right
5		Axis3+	up
6		Axis3-	down
7		servo rotate clock-wise	servo rotate clock-wise
8		servo rotate counter clock-wise	servo rotate counter clock-wise
9		screw grab	screw grab
10		screw release	screw release

If you have any question, please contact us on our official webpage: <http://www.dobot.cc>